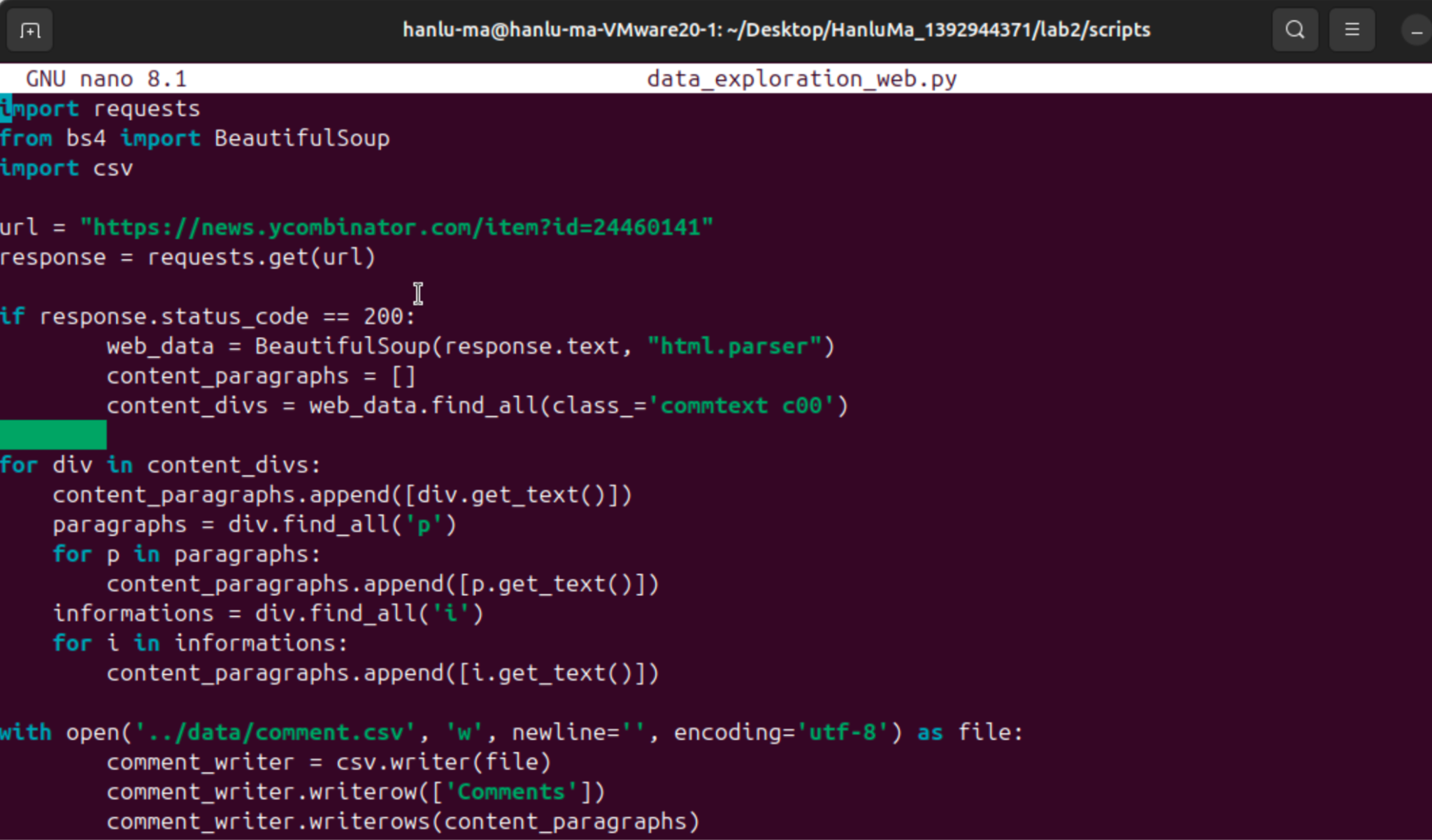
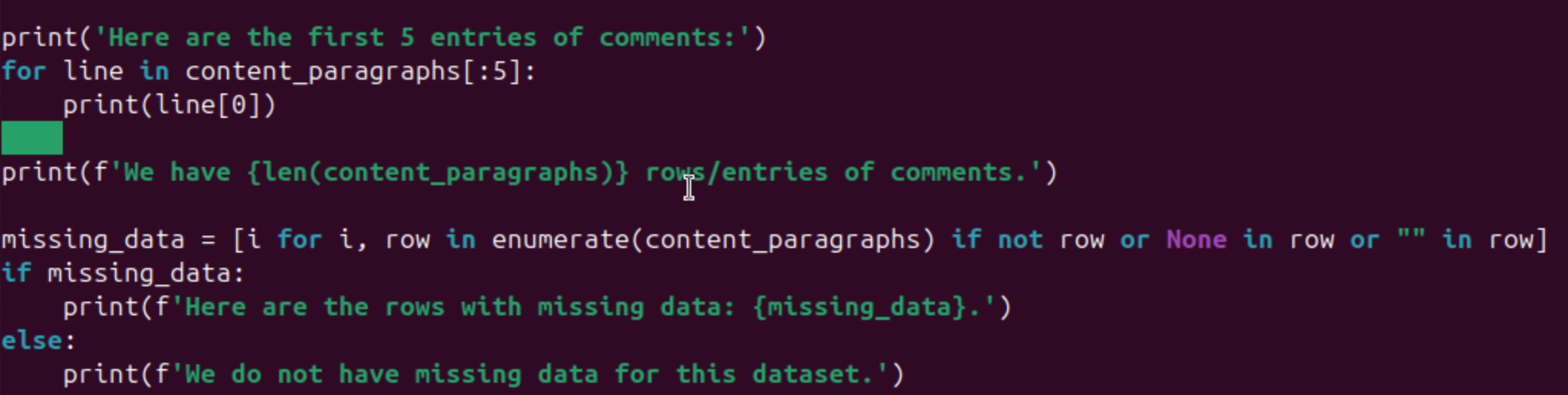
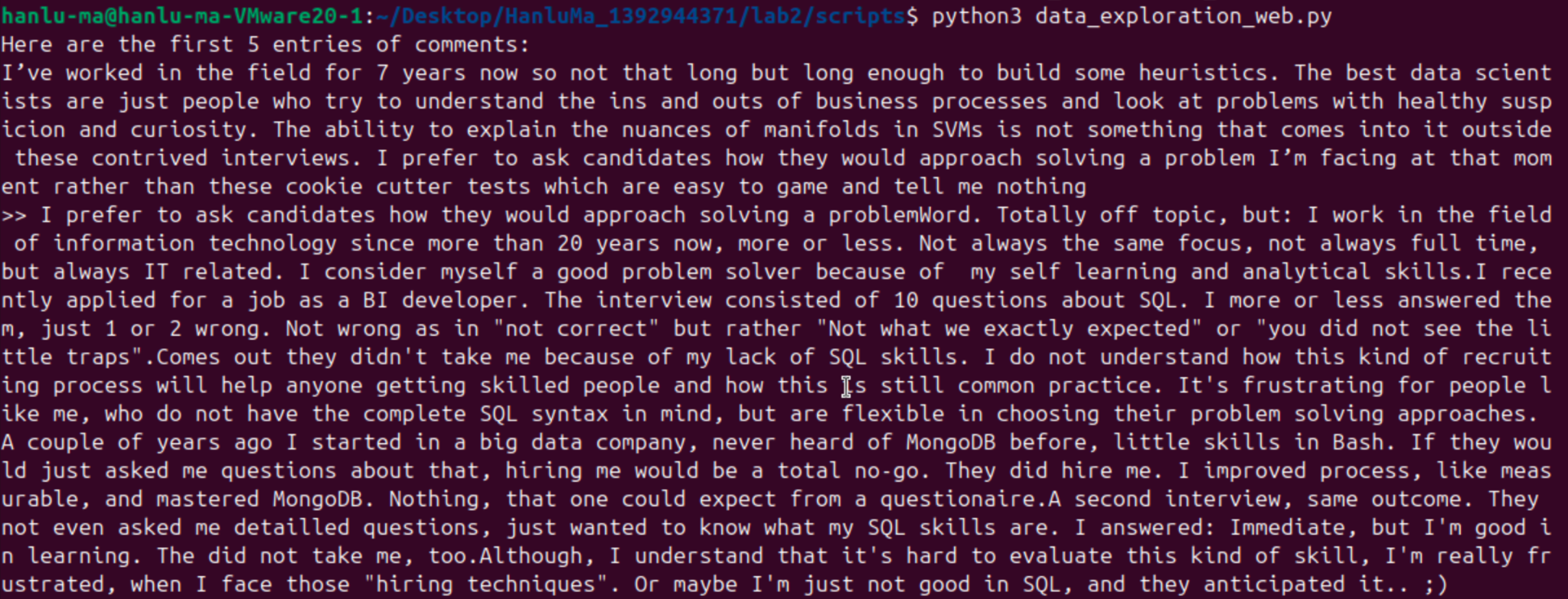
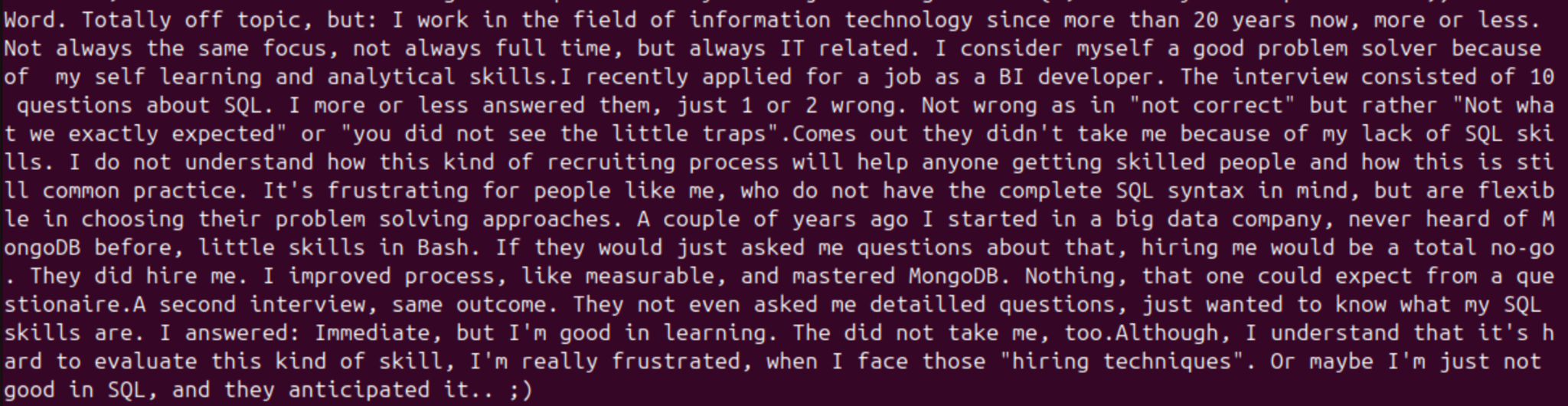
Data Science Professional Practicum (DSCI 560)

Laboratory Assignment 2

1. Team Formation
   1. Team name:
   2. Team numbers and their USC IDs:
      1. Hanlu Ma (USC ID: 1392-9443-71)
      2. Zhenyu Chen (USC ID: 2242-3773-15)
2. Data selection, Search, Find, and Collect
   1. Domain: DS job seeking and preparation content
   2. Reason:
   3. Dataset links and descriptions
      1. ASCII Texts like Forum Postings and HTML
         1. Link: <https://news.ycombinator.com/item?id=24460141>
         2. Description: this webpage entails peoples’ responses about how to prepare for data scientist interviews. After the data exploration phase, this dataset will provide some ‘empirical’ and ‘humane’ advice and suggestions about data scientist interview preparation.
      2. PDF and Word Documents that require conversion and OCR
         1. Link: <https://www.kaggle.com/discussions/general/177093>
         2. Description: the pdf file is obtained from the Kaggle website. It contains a comprehensive collection of Data Science interview questions that candidates may encounter during their interviews in the future, helping them better prepare.
3. Data collection
   1. CSV or Excel
   2. ASCII Texts like Forum Postings and HTML
      1. Code  
           
         
      2. Output  
         For data exploration, we generate the first 5 entries of comments, the number of comments/rows/entries, and the missing data situation to check the validity of the dataset.  
           
           
         
      3. Rationale of Code
         1. We use the ‘get’ method from the requests library to get the link and utilize ‘BeautifulSoup’ to obtain the information. Then we locate the html location of the context we want to parse, which is ‘commtext c00’ in this case. After deep diving into the webpage html, we notice that some comments go along with the div, but others appear with ‘p’ or ‘i’ tag under the ‘commtext c00’ div. Therefore, we use the ‘final\_all’ method and the ‘get\_text’ method to both obtain information along with the div and within the category of the corresponding div. After obtaining the relevant information and storing in the relevant variable ‘content\_paragraphs’, we write it into csv under the column name of ‘Comments’. Lastly, we have few lines to do some sanity checks about the dataset in terms of dataset length, missing values, and sample data extraction.
   3. PDF and Word Documents that require conversion and OCR
      1. Code

A computer screen shot of text

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

* + 1. Output

A screenshot of a computer program

AI-generated content may be incorrect.

* + 1. Rationale of Code
       1. First, we use the requests library to save the PDF file locally from the Kaggle webpage. Then, we use the “extract\_words(extra\_attrs=[‘fontname’, ‘size’])” method from the pdfplumber library to extract words from the PDF pages along with extra information – the font name and size of each word. After analyzing the font properties, we found that the font size of the section titles is 18, the font size of the question titles is 13, and the answer texts are size 12. Therefore, we distinguish the section lines, question title lines, and answers by comparing the current word's font size and the next word's font size. For instance, if the current word’s font size is greater than 15 and the next font size is between 13 and 15, then we start a new line between these two words. With this method, we are able to organize the content into three different lines. Moreover, for the answers' line breaking, we apply the re library to check if the word is followed by common punctuation symbols (“,”, “?”, “!”, and “:”). After we have processed the entire content in this format, we save the data into a text file locally.